

Amendments to the Drawings:

The Examiner rejected FIG. 1 because FIG. 1 should be “designated by a legend such as - Prior Art -- because only that which is old is illustrated.” As such, the appropriate legend has been added to the replacement figure that accompanies this amendment and response.

The Examiner rejected FIG. 8 because FIG. 8 should be “designated by a legend such as -Prior Art-- because only that which is old is illustrated. Applicant respectfully asserts that the figure does not represent “prior art.” Rather, FIG. 8 represents an exemplary implementation of the invention. The replacement paragraph provided in the amendments to the specification changes the permissive language in the specification to describe certain embodiments of the exemplary computer system that include the present invention. In addition, FIG. 8 has been amended to include components of the present invention in the exemplary embodiment of the computer system as described in the specification. In particular, the parenthetical “(Compiler, Constant Return Optimizer, or Control Operation Optimizer)” was added to the Operating System block 35. The amendment indicates, that in one embodiment, the compiler, constant return optimizer, and/or control operation optimizer is incorporated into the operating system. In addition, the parenthetical “(Constant Tables, vtables, Program Code)” was added to the Program Data block 38. The amendment indicates, that in one embodiment, the constant tables, vtables, and program code is incorporated into the program data. These changes obviate the objection and clearly show that FIG. 8 is not prior art. The changes to the specification and drawings do not add new matter but change the description and drawings to a better form.

REMARKS/ARGUMENTS

This Amendment and the following remarks are intended to fully respond to the Office Action dated February 10, 2005. In that Office Action, claims 1-23 were examined, and all claims were rejected. More specifically, claim 12 was objected to because of an informality; claims 22-23 were rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter; claims 1-23 were rejected under 35 U.S.C. § 103(a) as being unpatentable over “Compiler Transformation for High-Performance Computing” by Bacon et al. in view of “Memoization in top-down parsing” by Johnson. Reconsideration of these rejections, as they might apply to the original and amended claims in view of these remarks, is respectfully requested.

In this Response, claims 1, 12, 22, and 23 have been amended; no claims have been canceled; and no new claims have been added.

Claim Objections

Claim 12 was objected to because of the following informality: The addition of the word “that” produces a slightly confusing clause in line 5 of the claim. In response to the objection, the first instance of the word “that” was deleted.

Claim Rejections – 35 U.S.C. § 101

Claims 22-23 were rejected under 35 U.S.C. § 101 because the claimed invention is directed to non-statutory subject matter. Applicants do not acquiesce to the rejection and reserve the right to appeal or traverse the rejection at a latter time.

That said, the claims were amended to claim a computer program product. Since computer program product claims are statutory subject matter per *In Re Beauregard*, 53 F.3d 1583, 1584 (Ct. Fed. Cl. 1995), and as noted by the Examiner, it is believed these claims are in allowable form.

Claim Rejections - 35 U.S.C. § 103

Claims 1-23 were rejected under 35 U.S.C. § 103(a) as being unpatentable over “Compiler Transformation for High-Performance Computing” by Bacon et al. in view of “Memoization in top-down parsing” by Johnson.

Applicants respectfully traverse the section 103 rejections and respectfully requests reconsideration in light of the above amendments. With respect to the amended claims, the claims should be allowed over the prior art because one or more of the requirements of a prima facie case of obviousness is absent. Indeed, such a prima facie case can only be met when **all** of the following requirements are met: (1) there must be some suggestion or motivation in the references themselves (or in the knowledge available to those skilled in the art) to combine the references; (2) there must be a reasonable expectation of success; and (3) the combined references must teach or suggest all the claim limitations. See MPEP §§ 706.02(j) and 2143. In this case, Bacon does not describe generating, before invoking a target method, a return constant table having an entry associated with a constant return value of the target method of the receiver object or generating, before invoking a target method, an optimized instruction in association with the call site to retrieve, without requiring a function call, via the return constant table the constant return value associated with the target method. Indeed, Bacon teaches away from the present invention as described below.

The amended claims relate to a method or apparatus for optimizing indirect method invocation at a call site. The call site is associated with a receiver object, and programmed to call a target method of a plurality of possible target methods that return constant values. The present invention obviates the need for a function call at these sites. The invention obviates the need for the function call by generating a return constant table, which has constants as table entries. The constants are returned based on specific calls. Next, during compile time and before invoking a function or target method, instructions that would normally call the function that generates the constant are replaced with an optimized instruction that merely retrieves the constant value. As a result of this generated “return constant table” and the “optimized instruction,” there is no need to call the function.

Bacon relates to a system or method that relates to the construction of a cache to store recent invocation results. Bacon does not create an optimized instruction that optimizes the function call before invoking the target method. Bacon relies on the function calls to generate a cache of recent results, in contrast to the present invention, which replaces the function calls without first invoking a target method. Indeed, the passage that the Examiner cites shows the contrast between Bacon and the present invention:

“Memoization is an optimization that is applied to side-effect procedures (that is, procedures that do not change the state of the program, also called referentially transparent). In such cases it is possible *to cache the results of recent invocations*. *When the procedure is called again* with the same arguments, the cached result is used instead of re-computing it ...”

Bacon optimizes the return of results from a procedure call only after the method is invoked, and thus, over a period of time, Bacon will provide an efficient and optimized system. The present invention optimizes indirect method invocation before any method is invoked, and thus, it provides a more immediate, optimized system for method invocation. Thus, Bacon does not teach or suggest the limitations of the amended independent claims.

Each of the independent claims recited in the present application, i.e., claims 1, 12, and 22, relate to the generation, before invoking a target method, of the return constant table and an optimized instruction that retrieves a constant return value, wherein the optimized instruction is used to obviate the need for the function call. Bacon, because it still requires a function call, does not teach or suggest the present invention.

Johnson does not satisfy the inadequacies of Bacon. Johnson relates to the process of generalizing, “Norvig’s application of memorization to top-down recognizers to yield terminating recognizers for left recursive grammars.” Page 416, section 6. Johnson also does not create, before invoking a target method, an optimized instruction that obviates the need for a function call. Instead, Johnson specifically states that, “if the memorized procedure has not been called with args before, it is necessary to call the unmemoized procedure...” Page 414, paragraph 3. Like Bacon, Johnson teaches away from the present invention in that it calls the target procedure before it can optimize the method call. Thus, Johnson does not teach or suggest any of the limitations of the amended independent claims.

“Developing a Tool for Memoizing Function in C++” to McNamee et al., hereinafter “McNamee,” does not satisfy the inadequacies of Bacon and Johnson. McNamee describes the

development of an automated memorization utility. See page 17, column 2, paragraph 1.

McNamee also does not create, before invoking a target method, an optimized instruction that obviates the need for a function call. Instead, McNamee specifically states that a, “new function should check to see if it has been called before with the same argument, return the previously calculated value if so, and if not, invoke the original function...” Page 17, column 2, paragraph 2. Like Bacon and Johnson, McNamee teaches away from the present invention because McNamee must first call the method before optimizing the instruction. Again, the McNamee system becomes optimized over a period of time as methods are invoked, but, unlike the present invention, does not provide a more immediate, optimized method invocation. Thus, McNamee does not teach or suggest any of the limitations of the amended independent claims.

The combination of Bacon, Johnson, and McNamee simply does not teach or suggest each of the elements of the claimed invention. Bacon, Johnson, and McNamee, whether alone or in combination, fail to disclose the generation of an optimized instruction before invoking a target method as recited in amended claims 1, 12, and 22. Given that these references, both alone and in combination, fail to disclose, teach, or suggest all the claim limitations, they cannot, as a matter of law, render the amended claims obvious. Reconsideration of the § 103(a) rejections is therefore respectfully requested.

Claims 2-11, 13-21, and 23 depend from these independent claims, and thus, the dependent claims should be allowed for at least the same reasons, namely that the combination of the cited references does not teach the present invention. Since the remarks above are believed to distinguish over the applied reference, any remaining arguments supporting the claim rejections are not acquiesced to because they are not addressed herein.

Conclusion


It is believed that no further fees are due with this Response. However, the Commissioner is hereby authorized to charge any deficiencies or credit any overpayment with respect to this patent application to deposit account number 13-2725.

In light of the above remarks and amendments, it is believed that the application is now in condition for allowance, and such action is respectfully requested. Should any additional issues need to be resolved, the Examiner is requested to telephone the undersigned to attempt to resolve those issues.

Respectfully submitted,

Dated: 4/20/05





Tadd F. Wilson, #54,544
MERCHANT & GOULD P.C.
P.O. Box 2903
Minneapolis, MN 55402-0903
303.357.1651